

## Desenvolvimento de Modelo Conceitual para um Sistema de Gestão de Biblioteca

### Contexto e Objetivo:

A criação de um modelo conceitual para um sistema de gestão de biblioteca é um passo fundamental para garantir a eficiência na administração e organização das informações relacionadas aos livros, autores, usuários e empréstimos. Este modelo deve capturar as principais entidades e os relacionamentos entre elas, fornecendo uma base sólida para a implementação em um banco de dados relacional. A seguir, detalham-se as principais entidades e os relacionamentos necessários para a construção deste modelo.

### Entidades e Atributos:

#### 1. Livros:

A entidade "Livros" é central para o sistema e deve incluir informações essenciais sobre cada livro. Cada livro é identificado por um número único, o ISBN, e possui atributos como título, subtítulo (opcional), ano de publicação, gênero e uma descrição breve. É importante considerar que um livro pode ter múltiplos autores.

#### 2. Autores:

A entidade "Autores" armazena dados sobre os escritores dos livros. Cada autor deve ter um identificador único, nome completo, data de nascimento e uma biografia resumida. Um autor pode ter contribuído para a escrita de vários livros, e um livro pode ter vários autores associados.

#### 3. Usuários:

Para gerenciar os empréstimos de livros, o sistema deve manter informações sobre os usuários da biblioteca. Cada usuário é identificado por um número único e possui atributos como nome completo, endereço, e-mail e número de telefone. Adicionalmente, o sistema deve categorizar os usuários em diferentes tipos de assinatura, como "Aluno", "Professor" ou "Funcionário", dependendo do perfil do usuário.

#### 4. Empréstimos:

A entidade "Empréstimos" registra os detalhes dos empréstimos de livros. Cada empréstimo tem um identificador único e inclui a data de empréstimo, a data prevista de devolução e a data real de devolução (se disponível). Cada registro de empréstimo deve estar associado a um livro específico e a um usuário específico.

Relacionamentos:

### 1. Livros e Autores:

É necessário estabelecer um relacionamento muitos-para-muitos entre as entidades "Livros" e "Autores", pois um livro pode ser escrito por vários autores e um autor pode ter escrito vários livros. Este relacionamento deve ser adequadamente modelado para refletir essa complexidade.

### 2. Livros e Empréstimos:

A relação entre "Livros" e "Empréstimos" deve ser de um-para-muitos, onde um livro pode estar associado a vários empréstimos ao longo do tempo. Cada empréstimo deve referenciar um livro específico, e é essencial registrar a data de cada empréstimo para controle e histórico.

### 3. Usuários e Empréstimos:

Da mesma forma, a relação entre "Usuários" e "Empréstimos" deve ser de um-para-muitos, onde um usuário pode realizar vários empréstimos, mas cada empréstimo está vinculado a um único usuário. É fundamental registrar as informações sobre quais livros foram emprestados a quais usuários e as datas associadas a esses empréstimos.

Aqui está uma lista de exercícios de consultas SQL para praticar com a massa de dados fornecida. Esses exercícios cobrem uma variedade de operações SQL, desde consultas básicas até junções e subconsultas.

Massa de dados

-- Inserção de livros

```
INSERT INTO livros (ISBN, titulo, subTitulo, anoPublicacao, genero, breve_descricao)
```

```
VALUES
```

```
(9780140449136, 'A Odisséia', 'Traduzido por: João Pereira', 800, 'Épico', 'A história épica de Odisseu e sua jornada de volta para casa.'),
```

```
(9780321125217, 'Clean Code', 'A Handbook of Agile Software Craftsmanship', 2008, 'Tecnologia', 'Conselhos e melhores práticas para escrever código limpo e manutenível.'),
```

```
(9780201616224, 'Design Patterns', 'Elements of Reusable Object-Oriented Software', 1994, 'Tecnologia', 'Padrões de design de software e suas aplicações em programação orientada a objetos.');
```

-- Inserção de autores

```
INSERT INTO autores (nomeCompleto, dtNasc, biografiaResumida)
```

```
VALUES
```

```
('Homero', '0800-01-01', 'Poeta grego da Antiguidade, autor da Ilíada e da Odisséia.'),
```

```
('Robert C. Martin', '1952-12-05', 'Engenheiro de software e autor conhecido por seu trabalho em princípios de design de software.'),
```

```
('Erich Gamma', '1960-03-22', 'Um dos autores do famoso livro "Design Patterns", professor e consultor em design de software.');
```

-- Inserção de livrosXautores

```
INSERT INTO livrosXautores (cod_livro, cod_autor)
```

```
VALUES
```

```
(1, 1), -- A Odisséia por Homero
```

```
(2, 2), -- Clean Code por Robert C. Martin
```

```
(3, 3); -- Design Patterns por Erich Gamma
```

-- Inserção de usuários

```
INSERT INTO usuarios (nomeCompleto, endereco, bairro, cidade, estado, assinatura)
```

```
VALUES
```

```
('Ana Silva', 'Rua das Flores, 123', 'Jardim Primavera', 'São Paulo', 'SP', 'Aluno'),
```

```
('João Souza', 'Avenida Central, 456', 'Centro', 'Rio de Janeiro', 'RJ', 'Professor'),
```

```
('Maria Oliveira', 'Praça da Liberdade, 789', 'Liberdade', 'Belo Horizonte', 'MG', 'Funcionario');
```

-- Inserção de usuariosXtelefones

```
INSERT INTO usuariosXtelefones (idUserio, telefone)
```

```
VALUES
```

```
(1, 11987654321),
```

```
(2, 21987654321),
```

```
(3, 31987654321);
```

-- Inserção de usuariosXemails

```
INSERT INTO usuariosXemails (idUserio, email)
```

VALUES

```
(1, 'ana.silva@example.com'),  
(2, 'joao.souza@example.com'),  
(3, 'maria.oliveira@example.com');
```

-- Inserção de empréstimos

INSERT INTO empréstimos (dtEmpréstimo, dtPrevistaDev, dtRealDev, idUsuario)

VALUES

```
('2024-08-01', '2024-08-15', NULL, 1),  
( '2024-08-05', '2024-08-20', NULL, 2),  
( '2024-08-10', '2024-08-25', NULL, 3);
```

-- Inserção de empréstados

INSERT INTO empréstados (idLivro, codEmpréstimo)

VALUES

```
(1, 1), -- A Odisséia emprestado no empréstimo 1  
(2, 2), -- Clean Code emprestado no empréstimo 2  
(3, 3); -- Design Patterns emprestado no empréstimo 3
```

1. Listar todos os livros disponíveis na biblioteca.
2. Encontrar todos os autores que têm mais de 50 anos.
3. Mostrar todos os usuários que têm assinatura de 'Professor'.
4. Listar todos os empréstimos que ainda não foram devolvidos.
5. Exibir todos os livros junto com os nomes dos autores que os escreveram.
6. Mostrar todos os empréstimos com o nome do usuário que fez o empréstimo e o título do livro emprestado.
7. Listar todos os usuários e seus números de telefone.
8. Mostrar todos os e-mails dos usuários que têm assinatura 'Funcionario'.
9. Encontrar todos os livros publicados após 2000 e ordená-los pelo ano de publicação.
10. Listar todos os autores cuja biografia contém a palavra 'design'.
11. Mostrar os detalhes dos empréstimos que foram feitos por usuários da cidade 'São Paulo'.

12. Contar o número de livros de cada gênero na biblioteca.
13. Encontrar o autor com o maior número de livros na biblioteca.
14. Mostrar todos os usuários que nunca fizeram um empréstimo.
15. Listar os livros que foram emprestados, mas ainda não foram devolvidos.